EXHIBIT G

EXHIBIT 2

JOHN FRANKOVICH, NV Bar #667
LEIGH GODDARD, NV Bar #6315
McDonald Carano Wilson LLP
100 West Liberty Street, 10th Floor
P.O. Box 2670
Reno, NV 89505-2670
Telephone: (775) 788-2000
Facsimile: (775) 788-2020
Email: jfrankovich@mcdonaldcarano.com
lgoddard@mcdonaldcarano.com

KEVIN JOHNSON (*pro hac vice*)
RAY ZADO (*pro hac vice*)
SAM STAKE (*pro hac vice*)
Quinn Emanuel Urquhart & Sullivan, LLP
555 Twin Dolphin Drive, 5th Floor
Redwood Shores, CA 94065
Telephone: (650) 801-5000
Facsimile: (650) 801-5100
Email: kevinjohnson@quinnemanuel.com
rayzado@quinnemanuel.com
samstake@quinnemanuel.com

Attorneys for Defendant salesforce.com, inc.

UNITED STATES DISTRICT COURT
DISTRICT OF NEVADA

APPLICATIONS IN INTERNET TIME, LLC, Plaintiff, **v.** SALESFORCE.COM, INC., Defendant.	No. 3:13-CV-00628-RCJ-VPC DECLARATION OF BENJAMIN B. BEDERSON

I, Benjamin B. Bederson, declare as follows:

1. I have been retained by Defendant salesforce.com, inc. ("Salesforce") as an expert witness in this case to provide my opinions on the meaning of the terms in U.S. Patent Nos. 7,356,482 (the "'482 Patent") and 8,484,111 (the "'111 Patent") (collectively the "patents-in-suit"), both titled "Integrated Change Management Unit," asserted in this litigation by Plaintiff Applications In Internet Time LLC ("AIT"), as these terms would have been understood by a person of ordinary skill in the art at the time of the invention. If called upon as a witness, I could competently testify to the truth of each statement herein.

2. In preparing this declaration, I have reviewed the patents-in-suit and their file histories, AIT's Opening Claim Construction Brief, the Declaration of Craig Rosenberg Regarding Claim Construction, and such other matters as I identify below.

3. I reserve the right to supplement or amend this declaration based on any new information received that is relevant to my opinions, including any Reply Claim Construction Brief by AIT and any declarations and opinions in support thereof.

I. QUALIFICATIONS

4. Since 1998, I have been a Professor of Computer Science at UMD, where I have joint appointments at the Institute for Advanced Computer Studies and the College of Information Studies. I am also Associate Provost of Learning Initiatives and Executive Director of the Teaching and Learning Transformation Center. I am a member and previous director of the HCIL, the oldest and one of the best known Human-Computer Interaction research groups in the country. From 2006-2014, I was also co-founder and Chief Scientist of Zumobi, Inc., a Seattle-based startup that is a publisher of content applications and advertising platforms for smartphones. I am also co-founder and co-director of the International Children's Digital Library ("ICDL"), a web site providing the world's largest collection of freely available online children's books from around the world with an interface aimed to make it easy for children and adults to search and read children's books online.

5. From 1995 to 1997, prior to becoming a Professor at UMD, I was an Assistant Professor in the Computer Science Department at University of New Mexico. From 1992 to 1994 I was a Research Scientist at Bell Communication Research. From 1993 to 1994 I was also a Visiting Research Scientist at New York University ("NYU"). From 1990 to 1992 I was a Research Scientist at Vision Applications, Inc. From 1988 to 1990 I was a Teaching Assistant at NYU.

6. In addition, since 1993 I have consulted for numerous companies in the area of user interfaces, including Microsoft, the Palo Alto Research Center, Sony, Lockheed Martin, and NASA Goddard Space Flight Center.

7. For more than 25 years, I have studied, designed, and worked in the field of computer science and human-computer interaction. My experience includes 25 years of teaching and research, with research interests in human- computer interaction and the software and technology underlying today's interactive computing systems. This includes the design and implementation of user interfaces on client-server systems for querying data systems.

8. At UMD, my research is in the area of Human-Computer Interaction ("HCI"), a field that relates to the development and understanding of computing systems to serve users' needs. Researchers in this field are focused on making universally usable, useful, efficient, and appealing systems to support people in their wide range of activities. My approach is to balance the development of innovative technology that serves people's practical needs. Example systems following this approach that I have built include PhotoMesa (software for end users to browse personal photos), DateLens (software for end users to use their mobile devices to efficiently access their calendar information), SpaceTree (software for end users to efficiently browse very large hierarchies), ICDL (as described above), and StoryKit (an iPhone app for children to create stories).

9. At Zumobi, I was responsible for investigating new software platforms and developing new user interface designs that provide efficient and engaging interfaces to permit end users to access a wide range of content on mobile platforms (including the iPhone and Android-

based devices). For example, I designed and implemented software called "Ziibii," a "river" of news for iPhone, software called "ZoomCanvas," a zoomable user interface for several iPhone apps, and iPhone apps including "Inside Xbox" for Microsoft and Snow Report for REI.

10. Beginning in the mid-1990's, I have been responsible for the design and implementation of numerous other web sites in addition to the ICDL. For example, I designed and built my own professional web site when I was an Assistant Professor of Computer Science at the University of New Mexico in 1995 and have continued to design, write the code for, and update both that site (which I moved to the University of Maryland in 1998, currently at http://www.cs.umd.edu/~bederson/) as well as numerous project web sites, such as Pad++, http://www.cs.umd.edu/hcil/pad++/.

11. More recently, I have worked on complex web "apps," some of which include search results and synchronous and asynchronous client-server communications. These include my current home page, and a system called "Q&A" (http://www.cs.umd.edu/hcil/qa/) which supports real-time support for classroom interactive response by students with collaborative annotation of each other's responses.

12. I hold a B.S., M.S., and Ph.D. in computer science. I also earned an undergraduate minor in electrical engineering. I received the Janet Fabri Memorial Award for Outstanding Doctoral Dissertation for my Ph.D. work in robotics and computer vision. I have combined my hardware and software skills throughout my career in Human-Computer Interaction research, building various interactive electrical and mechanical systems that couple with software to provide an innovative user experience.

13. My work has been published extensively in more than 140 technical publications, and I have given approximately 100 invited talks, including 7 keynote lectures. I have won a number of awards including the Brian Shackel Award for "outstanding contribution with international impact in the field of HCI" in 2007, and the Social Impact Award in 2010 from Association for Computing Machinery's ("ACM") Special Interest Group on Computer Human Interaction ("SIGCHI"). ACM is the primary international professional community of computer

scientists, and SIGCHI is the primary international professional Human-Computer Interaction community. I have been honored by both professional organizations. I am an "ACM Distinguished Scientist," which "recognizes those ACM members with at least 15 years of professional experience and 5 years of continuous Professional Membership who have achieved significant accomplishments or have made a significant impact on the computing field." I am a member of the "CHI Academy," which is described as follows: "The CHI Academy is an honorary group of individuals who have made substantial contributions to the field of human-computer interaction. These are the principal leaders of the field, whose efforts have shaped the disciplines and/or industry, and led the research and/or innovation in human-computer interaction." The criteria for election to the CHI Academy are: (1) cumulative contributions to the field; (2) impact on the field through development of new research directions and/or innovations; and (3) influence on the work of others."

14. I have designed, programmed and publicly deployed dozens of user-facing software products that have cumulatively had millions of users. My work is cited in significant patents that are central to several major companies' user interfaces, including Sony and Apple.

15. I am the co-inventor of 8 U.S. patents, listed below. The patents are generally directed to user interfaces/experience.

- SanGiovanni, J., Bederson, B. (2014). Systems, Methods, and Computer Program Products Displaying Interactive Elements on a Canvas. US Patent #8,819,570.
- Pahud, M., Murillo, O. E., Karlson, A. K., & Bederson, B. B. (2012). Monitoring Pointer Trajectory and Modifying Display Interface. US Patent # 8,261,211.
- Good, L.E., Bederson, B. B., & Stefik, M.J. (2010). Methods and Systems for Supporting Presentation Tools Using Zoomable User Interfaces. US Patent # 7,707,503.
- Bederson, B. B., Good, L. E., & Stefik, M.J. (2010). Methods and Systems for Incrementally Changing Text Representation. US Patent # 7,650,562.
- Bederson, B. B., Good, L. E., & Stefik, M. J. (2009). Methods and Systems for Incrementally Changing Text Representation. US Patent # 7,549,114.
- Wallace, R. S., Bederson, B. B., & Schwartz, E. L. (1997). TV Picture Compression and Expansion. US Patent # 5,642,167.

- Bederson, B. B., Wallace, R. S., & Schwartz, E. L. (1993). Two-Dimensional Pointing Motor. US Patent # 5,204,573.
- Wallace, R. S., Bederson, B. B., & Schwartz, E. L. (1992). Telephone Line Picture Transmission. US Patent # 5,175,617.

16. I am being paid for my time in connection with this matter at my standard consulting rate, which is $600.00 per hour. My compensation is not dependent on the substance of my opinions, my testimony, or the outcome of this litigation.

17. My curriculum vitae, which includes a more detailed summary of my background, experience, and publications, is attached as Appendix 1.

II. OVERVIEW OF THE PATENTS

18. I have reviewed the patents-in-suit and their file histories. The patent are titled "Integrated Change Management Unit." I understand that the '482 patent issued on April 8, 2008 from U.S. Patent App. 09/797,488, filed on March 1, 2001, and that the '111 patents issued on July 9, 2013 from U.S. Patent App. 12/912,375, filed on October 26, 2010. I understand that AIT contends that both patents are entitled to claim priority to U.S. Patent App. No. 09/215,898, filed on December 18, 1998.

19. In my view, the patents-in-suit generally relate to their "Field of the Invention": "This invention relates to the integrated management of information affected by regulatory changes, such as changes in environmental, health and safety laws, and non-regulatory changes." (Col. 1:6-9.)[1]

20. I understand that to determine the ordinary and customary meaning of a claim term, one looks to the meaning that a person of ordinary skill in the art would have given the term at the time of the invention. I understand that AIT asserts that each asserted claims of the patents-in-suit has a priority date no later than December 18, 1998, the filing date of patent application 09/215,898. I further understand that AIT contends that conception of the patented inventions

[1] All citations to the patents-in-suit are with reference to the '482 Patent, unless I indicate otherwise.

occurred no later than December 1997, with a reduction to practice no later than June or July 1998.

21. Based on experience and the materials I have reviewed, it is my opinion that one of ordinary skill in the art for the asserted patents would have had the equivalent of a Bachelor's degree in computer science with two years of work or research experience relating to software for data processing and analysis functions.

A. **Claims**

22. I understand that AIT currently asserts claims 1, 3, 5, 6, 10, 20, 21, 23, 24,25, 26, 30, and 40 of the '482 patent and claims 13, 14, ,15, 16, 17 of the '111 patent. Claims 1 and 21 of the '482 patent, and claim 13 of the '111 patent, are independent claims. The remaining asserted claims are dependent claims.

23. These asserted independent claims require automatic detection of changes affecting a business:

- "the fourth portion of the server being configured to ***automatically detect changes that affect the information*** in the first portion of the server or the information in the second portion of the server." (111 patent, claim 13)
- "a change management layer for ***automatically detecting changes that affect an application***, each client computer further comprising a browser application being executed by each client computer, wherein a user interface and functionality for the particular application is distributed to the browser application and dynamically generated when the client computer connects to the server computer." (482 patent, claim 1)
- "***automatically detecting changes that affect*** a particular application" (482 patent, claim 21)

24. The asserted independent claims also require "dynamically generating" the functionality of and user interface for a business application in a manner that takes into account these external changes that result in corresponding updates to the claimed data and/or metadata:

- the third portion of the server being configured to dynamically generate a functionality and a user interface for the particular application, the functionality and the user interface of the particular application being based on the information in the first portion of the server and the information in the second portion of the server" ('111 patent, claim 13)

- a third layer associated with the server computer that retrieves the data in the first and second layers in order to generate the functionality and user interface elements of the application ('482 patent, claim 1)

- providing a third layer that retrieves the data in the first and second layers in order to generate the functionality and user interface for a particular application for the client computer as the client computer connects to the server computer ('482 patent, claim 21)

B. **Specification**

25. The '482 and '111 patents contain substantially identical specifications. The patents are both directed to "[a]n integrated system for managing changes in regulatory and non-regulatory requirements for business activities at an industrial or commercial facility." Abstract at 1-3.

26. The "Background of the Invention" describes the business challenges presented by frequent changes to regulatory, technological, and social requirements:

> [L]aws statutes, ordinances, regulations and related constraints are constantly changing and require corresponding changes in data entry, data analysis and presentation of the results." Col. 2:21-24.

27. The patents devote nearly one quarter of the specification—eight columns—to describing categories of often changing regulations and other business requirements that are the subject of the claimed invention. Col. 1:13-8:8. The patents categorize these changing requirements as regulatory, technological, or social changes:

> The system allows a business to use the normal business skills of their employees and does not require that every employee become a programmer in order to continue to respond to **regulatory and/or technological and/or social changes affecting business operations and/or information management requirements**.

Col. 22:34-39.

28. The specification provides examples of each type of requirement:

- **Regulatory Requirements.** The Toxic Substances Control Act regulates the generation, handling, and disposal of toxic and other hazardous substances, and regulatory reporting that documents the same. Col. 3:17-4:6.

- **Technological Requirements.** The Clean Air Act regulates vehicle and machine emissions of pollutants into the ambient air and atmosphere. Col. 2:51-67.

- **Social Requirements.** The Occupational Safety and Health Act regulates employee health and safety by ensuring minimal working conditions. Cols. 4:7-37.

29. These changes are not internal to or controlled by the business itself, but are published in third party sources that are made available via the Internet. *See* Abstract, ("The system … implements receipt of change information … using the facilities of the Internet."); see also Col. 10:21-28 ("Regulations and technical requirements are constantly changing in the United States. Regulatory changes are recorded and posted for reference in different media, including paper, microfiche and electronic media. The internet is one source of information on regulatory change that is both prompt and cost-effective."); Col. 10:31-42, 50-53.

30. The problem described by the patents-in-suit was how to update a business's software to reflect this "constantly changing" business landscape in the regulatory space. *See* Col. 2:21-24; 8:66-9:6. As noted in the patents, "[w]ithout an integrated method for automatically handling such changes, a developer or user of software that tracks business operations must continually rewrite part or all of the software in order to accurately and fully reflect these changes, usually at great expense and effort and with little hope for relief." Col. 9:4-9. The specifications note that the use of "regulations databases, document management systems and other partial solutions for tracking changes in, and compliance with, regulations and similar requirements," was well known in the art as of the time of the patents-in-suit. However, these systems purportedly did not provide an "integrated," or "closed loop" system that "identifies changes using intelligent network agents … and automatically effect(s) modifications in the system without the use of programmers and/or programming." Col. 7:56-67.

31. To address this problem of oft-changing regulations, the patents-in-suit describe as the "invention" an "integrated system for managing data [that] monitors, responds to, and incorporates changes in, federal, state and local laws, statutes, ordinances and regulations (referred to collectively herein as "regulations") and changes in technology in one or more regulated areas of commercial activity, such as environmental health and safety (EH&S), and food, drugs, cosmetics, medical devices and treatments ("FDCMTD")." Col. 9:10-16.

32. The purported invention thus "provides a relatively seamless system for creating robust solutions without the use of programmers and/or programming, (2) monitoring and

assimilating business change into business solutions rapidly, without (re)programming, and (3) providing business solution customization and extensibility without impacting the integrity or security of the system." Col. 9:26-32; see Col. 8:30-46. In so doing, the system avoids requiring developers or users to track these changes in voluminous regulations and manually update their databases and/or business applications to reflect such changes. See Col. 8:1-8.

33. The patents-in-suit further define the invention as including software components referred to as "intelligent agents." As stated in the patents: "[t]he invention begins tracking change using one or more intelligent agents ("IA's"), which are "specialized program[s] that reside[] on a network, or at a server as an applet, and can make decisions and perform tasks based on pre-defined rules." Col. 10:41-49. These intelligent agents "'cruise the Web' and identify and bring to the user's attention relevant regulatory and non-regulatory changes found on the Web that may affect a user's business." Col. 9:33-40; Fig. 1 ("Changes are identified on the internet using intelligent agents and provided for configuration.").

34. Figure 1 "schematically illustrates the relationship of four layers that are the primary components of the invention" Col. 8:50-51:

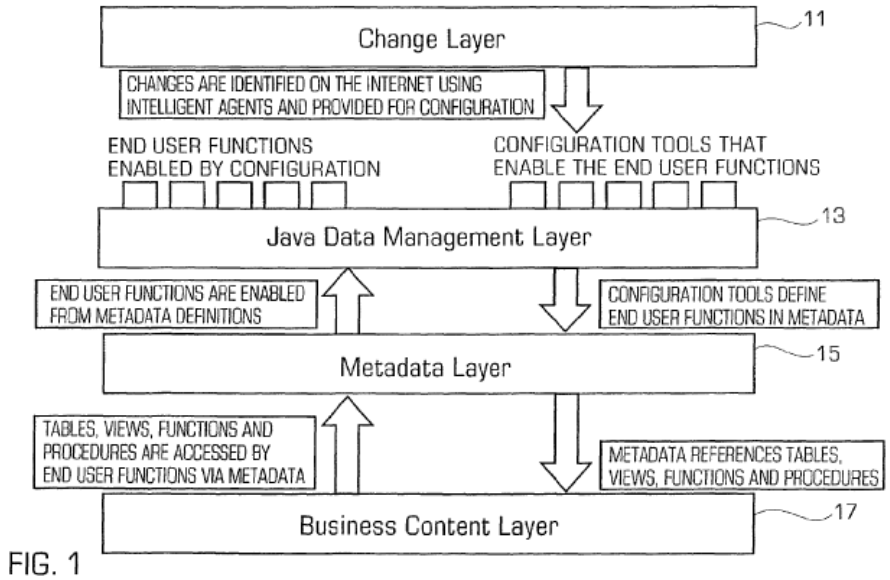


FIG. 1

35. As shown in Fig. 1 and set forth in the Specification, the invention operates in four distinct layers that comprise the "closed loop" system. The "change layer" or "change

management layer" (11 in Fig. 1 above) includes one or more intelligent agents that "'cruise the Web' and identify and bring to the user's attention relevant regulatory and non-regulatory changes found on the Web that may affect a user's business." Col. 9:34-38; Col. 16:18-34.

36. The intelligent agents next deliver any discovered changes to a "Java data management layer" (13 in Fig. 1 above). Col. 16:24-30. This Java data management layer "applies metadata attributes to business and business-change related data (regulation-based or non-regulation-based)." Col. 9:38-41; Col. 15:5-9. Importantly, the Java data management layer configures the changes from these third-party sources that are detected by the change management layer into end user functions to be incorporated into the business's software application, without the need for human intervention. Col. 9:49-52 ("Within the Java management layer, configuration tools take the place of a programmer and define various end user functions in terms of metadata, and metadata definitions are used to implement the desired end user functions."); Col. 16:24-30 ("A user may configure the system to apply pre-defined rules to the change in order to determine whether the change information delivered by the IA will be accepted and acted upon by the Java data management layer.").

37. The "metadata layer" "provides and/or defines data about every feature of the user interface [of an application] including, without limitation, tools, worklists, data entry forms, reports, … and other structures and functions." Col. 9:41-46.

38. Finally, the "business content layer" includes content "associated with a selected area of business activity." Col. 12:15-29; Col. 9:46-48. Figure 2 similarly depicts a "flowchart illustrating use of the invention to respond to one or more relevant changes found by an intelligent agent on a network. As set forth therein, the invention first detects changes on the Internet using intelligent agents ("IA"), and then configures "end-user functions" for incorporation into the application

C. **Prosecution History**

39. I understand that the '482 patent issued from U.S. Patent Application No. 09/797,488 (the "'488 Application"), which was filed on March 1, 2001. Ex. 2 at 1 ('482

Patent). In a January 18, 2006 Office Action, the Examiner rejected over 60 pending claims as anticipated by U.S. Patent No. 5,960,200 (the "Eager" reference).

40. The applicants responded to this Office Action on May 18, 2006. In this response, applicants sought to overcome the rejection in view of Eager by arguing that the Examiner improperly relied on the same group of functionality (labeled 130) to meet both the claimed change detection and change incorporation layers:

> How can Eager's functionality later 130 be both "a third layer associated with the server computer that retrieves the data in the first and second layers in order to generate the functionality and user interface elements of the application" and "a change management layer for automatically detecting changes that affect an application"? **At most, it can be one of those layers, not both of them together.**

Ex. 5 at 14 ('482 Patent File History, 5/18/2006 Amendment and Remarks (emphasis original)).

41. I understand that the applicants further attempted to overcome the Eager reference by amending the pending independent claims to explicitly incorporate the limitation of a "change management layer for automatically detecting changes." Id. at 2, 6, 13.

42. On February 28, 2007, the Examiner issued a Final Rejection, again rejecting all of the application's claims as anticipated by the Eager Reference. Ex. 7 at 2-6 ('482 Patent File History, 02/28/2007 Final Rejection). In response, on August 28, 2007, the applicants filed their appeal brief; in attempting to overcome the Eager reference, the applicants argued that human intervention, such as modification of application screens and messages as taught by Eager, is inconsistent with the claimed automatic change detection:

> Further, the Appellants note that Eager explicitly teaches that in the reengineering system, **it is "application developers and maintenance personnel" that "modify application screens and messages", thus teaching away from any means that would operate "automatically",** and in particular that would allow 'automatically detecting changes that affect an application' as recited in claim 2.

43. Ex. 8 at 14 ('482 Patent File History, 08/28/2007 Appeal Brief (emphasis in original)). The applicants emphasized that the claimed invention can generate a user interface "without requiring (re)programming of underlying software" and can incorporate detected changes "without requiring the services of one or more programmers to re-program and/or recode the

software items affected by the change." *Id.* at 2-3. I discuss the details of the Eager reference further below.

44. On December 28, 2007, the Examiner issued a Notice of Allowance. Ex. 9 at 1 ('482 Patent File History, 12/28/2007 Notice of Allowance).

D. **Specific Rebuttal To AIT and Its Expert's Characterization of the Patents-in-Suit**

45. In reviewing the declaration of AIT's expert, Mr. Rosenberg states in paragraph 6 that the patents-in-suit relate to "enabling individuals with knowledge of business processes, rather than only computer programmers, to have responsibility for application development with a simple and efficient metadata-driven application platform." I disagree, and find this to be an inaccurate reflection of the disclosure and claims of the patents-in-suit. Rather, as I state above, the patents-in-suit generally relate to their "Field of the Invention": "This invention relates to the integrated management of information affected by regulatory changes, such as changes in environmental, health and safety laws, and non-regulatory changes."

46. I further note that Mr. Rosenberg provides no explicit support for this assertion that the patents are directed specifically to "individuals with knowledge of business processes, rather than only computer programmers." Rather, the specification makes clear that the patents are directed to any organization where systems and methods for managing information affected by, e.g., regulatory changes are of importance. *See* Col. 8:9-26, 8:30-46.

47. In Paragraph 15 of his declaration, Mr. Rosenberg states that "The '482 and '111 patents discuss, as an example, a situation where changes in regulatory requirements may result in business changes in specific industries and consequently causing business applications to implement functional or data changes. Other types of changes, such as bug fixes and new features, may also result in modifications or updates to an application." Mr. Rosenberg provides no support for his characterization of that the patented invention as addressing modifying or updating an application with "bug fixes and new features." While certain aspects of the disclosure discuss reduced debugging time on account of the use of an object oriented language (Col. 14:26-33), and other aspects of the disclosure discuss, *e.g.*, creation and modification of forms and

reports (Col. 16: 35-47, 19:20-46), these are standard functionalities incorporated into business software of the time. However, the patents-in-suit focus particularly on the detection of changes to regulatory, technologic, and social requirements that affect a business. Neither AIT nor Mr. Rosenberg identify any other categories of changes disclosed in the specification as addressed by the patented invention.

48. In paragraph 17, Mr. Rosenberg asserts that the patents "describe a system where four different layers work in conjunction to allow users to easily modify an application or a set of applications to suit the users' needs without having to modify the applications source code (Change Layer, Java Data Management Layer, Metadata Layer, Business Content Layer). While I agree that the disclosed system includes four distinct layers, i.e., the Change Layer, Java Data Management Layer, Metadata Layer, Business Content Layer, which the patents identify as the "primary components of the invention," I do not find Mr. Rosenberg's comment to be an accurate characterization of the four layer architecture of the disclosed system. Rather, the four layer architecture that is referred to as the "primary components of the invention" comprise the "integrated solution" that allows for automatic detection and incorporation of regulatory, technological, and social changes without the need for human intervention.

49. In paragraph 18, Mr. Rosenberg claims that the metadata layer includes "a data dictionary and two types of metadata." He does not clarify what he means by this statement, but I find no support for it in the specification. Rather, as Mr. Rosenberg himself notes in paragraph 20 of his declaration, the description of the metadata layer simply provides that "[t]he metadata model has two main components, a business content data dictionary and an application component." Col. 12:32-41.

50. I do not disagree with Mr. Rosenberg's definition of "metadata" in paragraph 19, although I think it is an overstatement to say that metadata is used to "define all aspects of an application." Further, to the extent that Mr. Rosenberg states that "Different metadata tables may be used to store different types of metadata," this contention is unclear, and Mr. Rosenberg does not specify what he means by different "types" of metadata.

51. As I describe in more detail below, Mr. Rosenberg's statement in paragraph 21 that "Thus, at a high level, the '482 and '111 patents describe metadata within two layers: one layer includes metadata that defines the unique aspects of an application; and the other layer includes metadata that defines aspects common to a variety of applications. They correspond to the first and second information, respectively, as recited in the asserted claims" is unsupported by the patents' disclosure, and relies on arbitrary and undefined "types" of metadata.

52. For the same reason, I find Mr. Rosenberg's statement in paragraph 22 that "the data dictionary works in concert with the two types of metadata (i.e. "unique" and "common") to fully specify an application or set of applications" to be incorrect, and not reflective of the patents disclosure..

53. In Paragraphs 23-24 of his declaration, Mr. Rosenberg describes a hypothetical "software application" without any citation to the specification, despite the specification's disclosure of use cases of the patented invention. *See, e.g.*, Col. 10:30-12:7. Mr. Rosenberg's hypothetical "software application" does not practice the patented invention because at least, critically, he does not identify a change detection component that detects regulatory, technological, or social material change in third party repositories. Rather, Mr. Rosenberg is generally describing a generic process for creating a business application that was well known at the time of the invention of the patents-in-suit, with the exception of his reliance on the arbitrary and undefined "types" of metadata that find no support in the specification.

54. I have reviewed AIT's opening brief, and in particular, the first approximately six pages of that brief that purport to provide a background to the patents-in-suit, as well as the apparently corresponding paragraphs in .in Mr. Rosenberg's declaration.

55. I found AIT's description to be deficient in several regards.

56. First, as I discuss above, AIT incorrectly asserts that the asserted claims are directed to "a computer software architecture that allows application design and deployment to be performed by individuals with knowledge of the business process requirements of the customer." (Br. 1-2.).

57. Rather, as the patents explain in the context of discussing the "invention," the claims of the patents-in-suit are directed to an "integrated system for managing data [that] monitors, responds to, and incorporates changes in, federal, state and local laws, statutes, ordinances and regulations."

58. Next, AIT infers—incorrectly—that the alleged changes that are the subject of the claims are changes to "metadata." Again, this mischaracterizes the scope of the claimed invention, as I discuss above.

59. AIT also asserts that "personnel with knowledge of the business process, rather than software developers, have control over the design and implementation of applications." However, not only is this not a part of the claimed invention, which precludes such human intervention in the claimed integrated change management process, this functionality is no different than the art cited during prosecution, including the Eager reference (see my discussion below), which AIT had to distinguish on the grounds that it did not "automatically detect" changes.

60. I also note that AIT's description of the specification and Fig. 1 omit any reference to the "change layer," which is the functionality that performs the claimed change detection.

61. Instead, based on my review, AIT focuses much of its background discussion on the business content layer and metadata layers, asserting that the "data elements of the business content layer include metadata." Br., p. 4. However, the patents' disclosure regarding the business content includes no reference to metadata. The only explicit reference in the specification to metadata that I noted in my review is in the metadata layer.

62. I further note that, in my review of Mr. Rosenberg's declaration, he apparently does not take the position that the business content layer includes metadata.

63. AIT also includes in its background discussion a description of Fig. 4A of the patents. Fig. 4A is a graphical representation that illustrates relationships between metadata tables in the metadata layer.

64. In paragraph 25 of his declaration, Mr. Rosenberg claims that "the metadata defining the common aspects of a variety of applications is described as standard interface system, or SIS. ('482 and '111 patents at Figs. 3-5.) The SIS tables define application functions that are used to generate various application components. These tables are part of the metadata layer. ('482 patent at 12:54-55. '111 patent at 12:58-59.) SIS has the advantage of not requiring (re)programming to respond to changes made to the applications because the system is dependent on and driven by metadata. ('482 patent at 11:64-12:2. '111 patent at 11:66-12:4.) I find this to be an inaccurate characterization of the disclosure of the patents-in-suit.

65. First, the "standard interface system" appears to be nothing more than a reference to the Java data management layer and the metadata layer: "The Java data management layer and the metadata layer together serve as a standard interface system that is positioned 'on top of' one or more databases, allowing addition, deletion and modification of data entry forms, tables, views, images, reports, queries, information processing and logic, monitoring or work flow and distribution and routing, menu presentations and provision of regulatory or non-regulatory alerts." Col. 16:4-16.

66. Next, Figs. 3-5 of the specification contain no reference to "common aspects of a variety of applications," as Mr. Rosenberg claims. Fig. 3 illustrates "structures and relationships" between various tables that are used to implement worklists, such as the business process table, the worklist table, and the module table. Col. 13 20-36. Fig. 4A is an expanded view of the relationships between these and other tables. Fig. 5 illustrates another set of tables related to business rules. Col. 13:65-14:19.

67. I also disagree with Mr. Rosenberg's statement in paragraph 26 that "the metadata that defines the application, including both the common and the unique metadata, is interpreted …." Again, there is no disclosure of such "common v. unique distinctions or types of metadata using these arbitrary and undefined categories. Notably, Mr. Rosenberg is inconsistent with and departs from the patents' disclosure and express definition in his reference to "data mapping," The "data mapping" disclosed in the patents is not based on "common and unique metadata," but more

simply on the mapping of an item in a GUI and an item to be changed in the business content later: "'Data mapping,' as used here, refers to a mechanism that provides a correspondence between an item in a graphical user interface (GUI) and an item to be changed in the business content layer, which is used as part of the generation of the user interface. Col. 15:17-25.

68. In paragraph 29, Mr. Rosenberg notes that "By using the metadata to define software applications, the invention is able to automate the software modification process by generating an application's executable code from interpreting its metadata. Moreover, since only the metadata needs to be modified to incorporate changes to an application, which eliminates the need to modify or rewrite the application's source code, a person without extensive software programming skill may also be able to modify the application's metadata and thus making changes to the application. This lessens the demand on software developers and programmers as well." To the extent he is referring to the claimed inventions of the patents-in-suit, this is not correct; rather, this is functionality that was well know in the art at the time (as I describe below). Further, and importantly, this functionality is inconsistent with the "automatic" detection of the claims, which precludes such user interaction, including rekeying and reformatting of forms (as I discuss below).

69. Returning to AIT's brief for the moment, I find its description to likewise mistakenly characterize Fig. 4A to the extent it claims that "[w]ithin the SIS tables, the metadata attributes are then designated as representing data elements that are either 'common' to various applications or 'unique' to a single application." Br. at 4-5. There is no disclosure of this alleged "common / unique" distinction in the '482 patent, including in connection with the tables of Fig. 4.

70. In its brief, AIT also addresses Figs. 6 and 7, which are not referenced in Mr. Rosenberg's declaration.

71. In my view, AIT is incorrect when it alleges that these tables "illustrate the difference between the innovative system claimed in the asserted patents." Critically, these tables do not relate to the "automatically detecting" changes that is the subject of the claims. Rather,

these tables relate to manually creating a data entry form – functionality that is the very opposite of "automatic." Col. 16, 35-36, 47-51.

72. Moreover, as disclosed in the patents-in-suit, this manual creation of data entry forms takes place *after* changes in regulatory requirement have been detected by intelligent agents (the operation that is the subject of the claimed invention).

73. This is functionality that is also provided by the Eager reference that I discuss in more detail below, which the applicants had to distinguish during prosecution..

74. Returning to paragraphs 30-32 of Mr. Rosenberg's declaration, Mr. Rosenberg departs from the express disclosure of the specifications of the patents-in-suit, and retroactively attempts to impose arbitrary distinctions on the metadata that is referenced in the context of the metadata layer. Notably, when referencing the alleged metadata "common to a variety of applications," he is unable to cite to the specification for this language, as it simply does not exist. He then cites alleged benefits that, again, are not present in the disclosure of the patens in suit.

III. LEGAL STANDARDS

A. Claim Construction

75. I understand that the claim language determines the scope of the invention. I further understand that the claims are interpreted in the context of the intrinsic record: the claim language, the specification and its figures, and the prosecution history. Descriptions of the invention in the specification give life and meaning to claim language. In particular, where the specification describes the features of the "invention" as a whole, this description limits the scope of the invention. I also understand that, when a patentee makes a clear and unmistakable disavowal of scope during prosecution, a claim's scope may be narrowed under the doctrine of prosecution disclaimer. Finally, I understand that extrinsic evidence is less reliable in construing claims, but can help educate the court regarding the field of the invention and can help the court determine what a person of ordinary skill in the art would understand claim terms to mean.

B. **Indefiniteness**

76. I understand that a patent is invalid for indefiniteness if its claims, read in light of the specification delineating the patent, and the prosecution history, fail to inform, with reasonable certainty, those skilled in the art about the scope of the invention. Further, the claims, when read in light of the specification and the prosecution history, must provide objective boundaries for those of skill in the art.

IV. DISPUTED CLAIM TERMS

A. "automatically detecting"

Term	Salesforce Proposed Construction	AIT Proposed Construction
"automatically detect[ing]" '482 Patent, claims 1, 21. '111 Patent, Claim 13.	Indefinite, or in the alternative, requiring at least "detecting without any intervention by a human operator through the use of one or more intelligent agents"	"detecting without direct human intervention"

77. Both Mr. Rosenberg and I appear to agree that "automatically detecting" changes requires that the system perform the claimed automatic detection of changes. Although not clear, we appear to disagree, however, as to whether "human interaction," indirect or otherwise, is involved in this automatic process.

78. In my opinion, one of ordinary skill in the art at the time of the claim invention would have understood that the plain meaning of "automatically" required detection of changes without any intervention by a human operator.

79. In my view, contemporary technical dictionaries confirm this understanding, e.g.,:

- "automatic" - a process or a device that functions ***without intervention by a human operator*** under specified conditions. Ex. 12 at 41 (Comprehensive Dictionary of Electrical Engineering 1999).
- "automatic" - 1. self-regulating or self-acting; capable of producing a desired response to certain predetermined conditions. 2. Self-acting and self-regulating;

operating without human intervention; often implying the presence of a feedback control system. 3. Pertaining to a process or device that, under specific conditions, performs its functions without intervention by a human operator. Ex. 13 at 66 (Modern Dictionary of Electronics (6th ed. 1997).

- "automation" - ***The replacement of human skill by automatic machine operations***. Word processing software is an example of the potential of automation. These programs automate tasks as simple as centering text and as complex as sorting a mailing list into zip code order. Ex. 14 at 47 (Webster's New World Dictionary of Computer Terms (2000).

80. I note that AIT cites just one dictionary definition, the first definition above, as supporting its proposed construction. However, this definition supports Salesforce's proposed construction, as it rules out all human intervention under the specified condition in which the automatic function or process operates.

81. In my view, the specification confirms this understanding of the plain meaning. Specifically, the patentee defines the "invention" as involving an "integrated," or "closed loop" system that "identifies changes using intelligent network agents … and automatically effect(s) modifications in the system without the use of programmers and/or programming." The specification further makes clear that this aspect of the "invention" is performed without human involvement: "the invention provides a relatively seamless system for creating robust solutions without the use of programmers and/or programming, (2) monitoring and assimilating business change into business solutions rapidly, without (re)programming," Col. 9:26-32, *see also* 8:30-46, 9:33-48, 9:64-10:3; Abstract at 6-19; 8:1-26.

82. As a result, my view is that the claims would be naturally and properly construed as directed to this integrated, closed loop system.

83. I note that the specification contrasts this functionality with certain activities that are characterized as "manual," all of which involve some degree of user involvement, including

entry of data into a data form (Col. 18:43-45), creating worklists (Col. 10:61-62), or creating processes (Col. 11:41-42).

84. Under AIT's proposed construction, these "manual" activities could all meet the "automatically detecting" limitation, as they could be said to comprise "indirect" human interaction, with the system performing the "detecting" of changes once the user had programmed the form, worklist, or report.

85. AIT also identifies two instances in the specification that purportedly envision *indirect* human intervention in an "automatic" process. (Br. at 10.) However, neither teaches or suggests any indirect human intervention. First, the specification refers to an "integrated method for automatically handling" changes, including "changes in regulations, in the business environment, in technology and in any other factor." Col. 8:67-9:9. This passage, however, does not disclose or imply any indirect human intervention with the "automatic detection of changes." Rather, all this section states is that without the "integrated method" of automatically handling changes in regulations, a developer or user of a system that tracks overall business operations needs to become involved in the regulatory change detection and implementation process. Indeed, this passage supports the view that the stated goal of the invention disclosed in the patents-in-suit is a truly automatic process.

86. Second, the specification states that "Configuration Users can choose to automatically configure the preceding recommendation based on a set of default conditions, or can manually implement the configuration using a configuration toolkit." Here, the specification does not state that indirect human involvement is involved in automatic configuration, but rather states that there is an alternative to automatic configuration: manual implementation. The specification later makes clear that this automatic configuration is fully automatic: "A user may configure the system to apply ***pre-defined rules*** to the change in order to determine whether the change information delivered by the IA will be ***accepted and acted upon*** by the Java data management layer."

87. AIT's suggestions that human operators may be indirectly involved in change detection not only has no support in the specification, but also the applicants' amendments and statements in the prosecution history weigh against it. More particularly, the patentee incorporated the term "automatically detecting" into the independent claims to overcome the prior art Eager reference, and specifically represented that user intervention, such as modification of application screens and messages, is not permitted in such automatic change detection:

> Further, the Appellants note that Eager explicitly teaches that in the reengineering system, **it is "application developers and maintenance personnel" that "modify application screens and messages", thus teaching away from any means that would operate "automatically",** and in particular that would allow 'automatically detecting changes that affect an application' as recited in claim 2.

Ex. 8 at 14 ('482 Patent File History, 08/28/2007 Appeal Brief.)

88. Eager disclosed a "re-engineering subsystem 30" that provided for "custom application development or re-engineering." Ex. 10, Col. 4:28-30. The application subsystem provided several tools for modifying graphical user interface, or GUI files, which could be viewed as a graphical user interface editor. *Id.*, Col. 23:66-24:11.

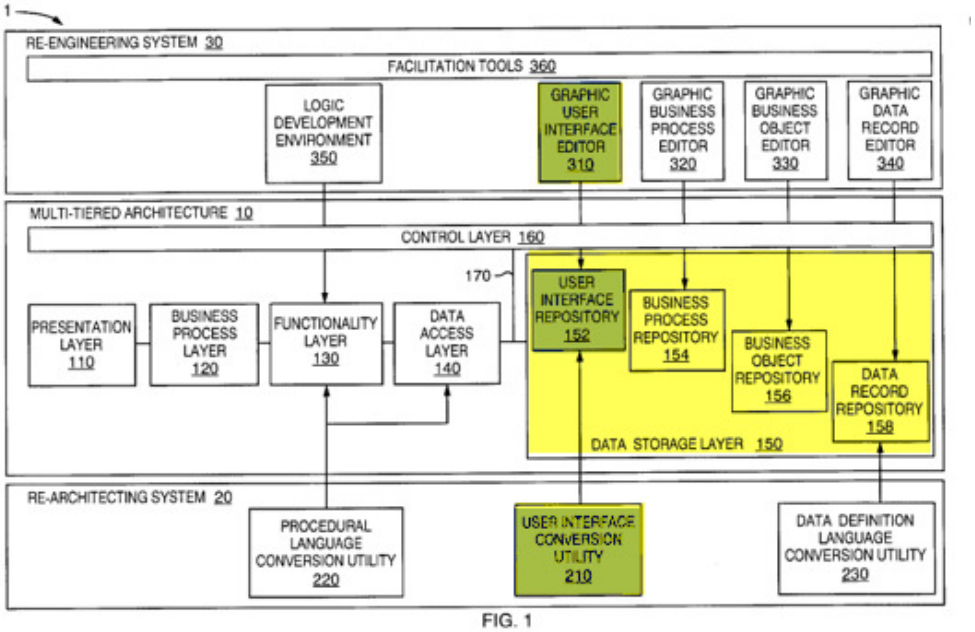


FIG. 1

89. The graphical user interface editor could then, based on a user positioning of representations of certain objects, called "business objects" on a screen, enable "a certain amount of the application code to be generated automatically from graphical representations," *i.e.*, the system would detect the changes to metadata resulting indirectly from the user's actions, and then generate code corresponding to those changes. *Id.*, Col. 24:12-20, Col. 30:45-64.

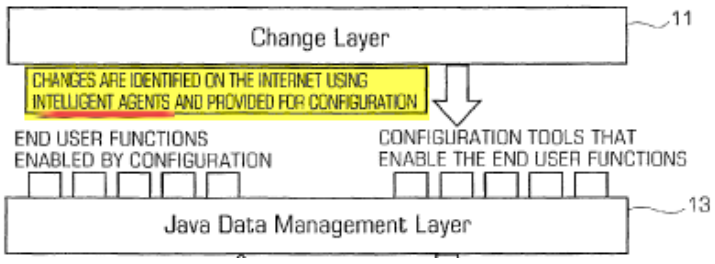
90. The applicants distinguished this functionality, which involved detecting modifications based on the user of the graphical user interface editor (as well as other editors), from the claimed "automatic detection of changes," as there was some user involvement. The user in this example was "indirectly involved" in the detection of changes to the metadata, as AIT would improperly include in its construction now.

91. Finally, in my view, AIT's proposed construction would render the claims indefinite, as the patents provide no "objective boundaries for those of skill in the art" as to the degree of human intervention permitted during change detection.

92. The specification further defines the "invention" in particular as using intelligent agents to perform the automatic detection.

93. In paragraph 37 of his declaration, Mr. Rosenberg attempts to dismiss the use of intelligent agents as "merely one embodiment of change detection." I disagree with Mr. Rosenberg for several reasons.

94. First, as I discuss above, the patents do not refer to use of intelligent agents as "an embodiment," but rather, as component of the "invention." This is confirmed by the depictions of the "invention" in Figures 1 and 2:



95. The specification further repeatedly describes intelligent agents as core component of the invention. In characterizing the invention as depicted in Figure 1. the specification states:

> The system operates at four layers, as illustrated in ***FIG. 1***: (1) a change management layer **11** that includes one or more change agents that "cruise the Web" and identify and bring to the user's attention relevant regulatory and non-regulatory changes found on the Web that may affect a user's business.

Col. 9:33-38.

96. In illustrating application of the invention in complying with toxic waste regulations, the specification confirms that intelligent agents are an integral component of to the claimed invention:

> The following example illustrates how a change, made to a regulation, is ***identified on the Internet and incorporated and managed by the invention***.
>
> . . .
>
> ***The invention begins tracking change using one or more intelligent agents ("IA's").*** An "intelligent agent" is a specialized program that resides on a network, or at a server as an applet, and can make decisions and perform tasks based on pre-defined rules. Preferably, two or more IA's used by a business will have sufficiently different assignments that at most modest overlap occurs between the IA's.

Col. 10:21-49.

97. The specification further describes the use of intelligent agents when it describes the Change Management Layer:

> E. Change Management Layer
>
> The change layer primarily involves an intranet or the Internet ***and uses one or more intelligent agents (IA's)*** that ***continually search on the Web*** for relevant changes in a selected business area. The changes may be regulatory and/or non-regulatory, and each IA is defined by rules and constraints that focus on the selected business area.

Col. 16:17-23.

98. Both Mr. Rosenberg in paragraph 37, and AIT in its brief, argue that there are other embodiments of the claimed "change detection" in the patents-in-suit. However, use of intelligent agents is the only mechanism for the claimed change detection of, *e.g.*, regulatory requirement disclosed or enabled by the specification.

99. Mr. Rosenberg refers to a set of "Change Configuration functions" that "support change of End User functions 'through a variety of flexible and intelligent manual routines, such as intelligent agents, screens, fields, reports, documents and logic that can be changed without requiring programming skills.'" However, these "intelligent manual routines," do not perform the claimed detecting of changes; rather after changes have already been detected, these manual routines may be used for certain "Change Configuration" functions:

> The invention provides a cost-effective approach for **absorbing database and application changes that arise from changes in regulations, policies, procedures, processes, materials, and similar factors**. The integrated framework of the invention is divided into two main groupings, Change Configuration functions and End User functions. The **Change Configuration functions support creation and change of End User functions** through a variety of flexible and intelligent manual routines, such as intelligent agents, screens, fields, reports, documents and logic that can be changed without requiring programming skills.

Col. 10:4-14.

100. I further note that these alleged other examples refer to "manual routines." In my view, these manual routines cannot refer to the claimed "automatic detection of changes": such manual operation is, on its face, the opposite of the claim requirement.

101. Moreover, these manual examples, including "screens, fields, reports, documents and logic" are not capable of "cruising the web" to locate changes in regulatory, social, or technological changes, as required by the claims.

B. **"changes that affect . . ." ('111 patent, claim 13, '482 patent, claims 1, 21)**

Term	Salesforce Proposed Construction	AIT Proposed Construction
"changes that affect the information in the first portion of the server or the information in the second portion of the server" '111 Patent, claim 13. "changes that affect a particular application" / "changes that affect an application"	"modifications to regulatory, technological, or social requirements stored in a third-party repository that affect information about unique aspects of a particular application or functions common to various applications" "modifications to regulatory, technological, or social requirements stored in a third-party repository that affect an application"	"changes to an application's metadata"

'482 Patent, claims 1, 21.		

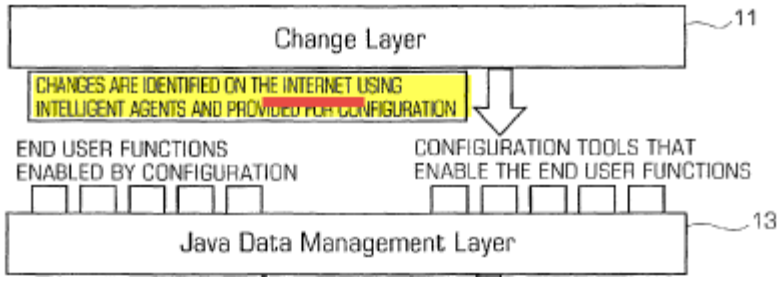
102. First, in my view the plain meaning of "affecting" suggests that the source of change is one step removed from the claimed system, not incorporated within it.

103. Thus, I disagree with Mr. Rosenberg's reading of the claim language in paragraph 44 of his declaration, where he states that "Since an application is generated using its corresponding metadata, any change to the metadata necessarily affects the application generated based on the metadata." Rather, the term "affect" the data appears to have been carefully chosen to embody the requirement that the detected regulatory, technological and/or social changes have to affect the business. If the patentee had intended to cover changes *to* the metadata, they could have simply used the word "to."

104. Moreover, I note that this reading is inconsistent with the specification's characterization of the invention, which involves changes on the Internet from third party repositories that affect a user's business. Col. 2:21-24; 8:66-9:6; 10:31-53.

105. Further, in my view, the specification makes clear that the claimed detected changes are in third party repositories, not within the system itself, such as is proposed in paragraphs 43-44 of Mr. Rosenberg's report.

106. The specification notes that the described (and claimed) system "receives information on regulatory and non-regulatory changes that affect operations of the business," thereby indicating hat the changes are sent from somewhere outside the system. Further, these changes are in fact detected by intelligent agents that "cruise [or search] the web" to identify such changes that affect a business:



Figs. 1-2; Col. 9:33-37; Col. 16:18-21 ("one or more change agents that 'cruise the Web'" and identify and bring to the user's attention relevant regulatory and non-regulatory changes found on the Web").

107. Although Mr. Rosenberg does not so state, AIT argues that there is no disclosure of any third party repositories in the specification. (Br. at 14.)

108. I disagree: the only sources of changes detected by the described and claimed "integrated system" are from third party repositories that are accessed "on the web" (*see* Col. 9:33-38) and via the Internet. For example:

> Regulations and technical requirements are constantly changing in the United States. Regulatory changes are recorded and posted for reference in different media, including paper, microfiche and electronic media. ***The internet is one source of information on regulatory changes that is both prompt and cost-effective.***

Col. 10:21-26.

109. In fact, AIT's proposed construction cannot work in the context of the sole embodiment of the invention, which the patentee equates to the invention. In the disclosure as reflected by Fig. 1, changes are detected by the change layer, which comprise one or more intelligent agents that search the Internet for changes. The change layer—which Mr. Rosenberg and AIT barely discuss—never communicates with the metadata layer, and thus cannot detect changes to metadata. Rather, the change layer passes on the changes it detects to the Java data management layer. *See* Fig. 1; Col. 16:18-34.

110. Moreover, based on my reading, the prosecution history is inconsistent with a proposed construction that the changes that are detected are changes in "metadata."

111. I note that during prosecution of the '482 patent, the applicants' filed an appeal brief dated August 28, 2007. In that brief, the applicants specifically identified the disclosure that corresponds to the claimed "change detection."

112. The applicant pointed to only two areas of the specification as related to the change detection claim limitations: box 21 of Fig. 2 (which reads "IA identifies one or more relevant changes on a network using pre-defined rules"), and page 28, lines 4-16 of the as-filed specification. This corresponds to Col. 16:18-30 of the issues patent:

> The change layer primarily involves an intranet or the Internet and uses one or more intelligent agents (IA's) that continually search on the Web for relevant changes in a selected business area. The changes may be regulatory and/or non-regulatory, and each IA is defined by rules and constraints that focus on the selected business area. When an IA discovers a relevant change, the IA obtains all available information concerning this change and delivers this information to the Java data management layer. A user may configure the system to apply pre-defined rules to the change in order to determine whether the change information delivered by the IA will be accepted and acted upon by the Java data management layer.

Ex. 11 at 28 ('482 Patent as-filed specification); *see* Ex. 8 at 4 ('482 Patent File History, 08/28/2007 Appeal Brief (cross-referencing in 08/28/2007 Appeal Brief)).

113. Neither of these citations refer in any way to detecting changes in "metadata"; rather, they each directly correspond to intelligent agents detecting relevant changes in third party repositories in accordance with Salesforce's proposed construction.

114. Furthermore, Figure 1 clearly shows an arrow flowing from the Change Layer (11) to the Java Data Management Layer (13), but there is no arrow flowing back to the Change Layer from the Java Data Management Layer (or the Metadata or Business Content Layers for that matter):

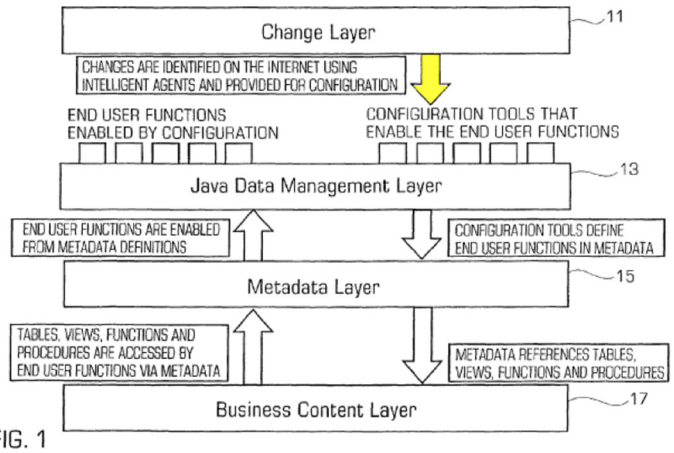


FIG. 1

115. Thus, it is clear that the patent does not provide any mechanism for the change layer to have access to the application's metadata. Thus the Change Layer could not possibly detect changes to the metadata since it can not access the metadata.

116. The specification also discloses that the claimed invention detects regulatory, technological, and social changes, as Salesforce proposes, not any and all changes regardless of their materiality or relation to a business. The claims require that such changes must "affect" the business with sufficient materiality to require the change to be incorporated into the application.

117. According to the specification, these changes include changes to statutes, laws, and ordinances ("regulatory changes"), changes to technology in regulated areas ("technological changes"), and changes to health and personnel requirements in regulated areas ("social changes"):

> The system allows a business to use the normal business skills of their employees and does not require that every employee become a programmer in order to continue to respond to **regulatory and/or technological and/or social changes affecting business operations and/or information management requirements**.

Col. 22:34-39; *see also* Col. 9:10-32.

118. Beyond these categories, I am not aware of any other material changes disclosed in the specification that are or can be detected by intelligent agents.

119. In my view, AIT's proposed construction—which points to the updating of "metadata" (presumably in the first or second layer / portion of the server) as the claimed detection of changes—is antithetical to the express purpose of the invention. Under AIT's proposed construction, a user could, in response to a regulatory change that the user manually detected by reviewing the federal register, reprogram the claimed first and second layers to reflect this change in regulation. However, so long as the system then "automatically" detected the updating of the manually changed metadata, the claimed limitation would purportedly be met. This construction, which defeats the purpose of the invention (i.e., preventing this very "rewrit[ing] part or all of the software in order to accurately and fully reflect these changes" (Col. 9:4-9)), appears to be incorrect.

120. Thus, AIT's proposed construction would render superfluous the claimed change management layer / fourth portion of the server. Specifically, the claim requires the dynamic generation of functionality and user interface based on information in the first and second layers. If the claimed change detection was based on detection of updates to the metadata in those first and second layers, there simply is no reason to have any separate change detection. Rather, the "changes" are taken into account whenever the functionality and user interface are dynamically generated. Furthermore, the change management layer simply doesn't have access to the metadata layer so it couldn't detect changes in the metadata. Instead, the change management layer/ fourth portion of the server is required because the relevant changes that are detected take place outside the claimed system.

121. Therefore, in my opinion, Salesforce's proposed construction is correct.

C. **Dynamic Generation**

Term	Salesforce Proposed Construction	AIT Proposed Construction
"dynamically generate a functionality and a user interface" '111 Patent, claim 13. "dynamicaly [re-]generate[d, ing]" '482 Patent, claims 1, 21.)	Indefinite, or in the alternative, requiring at least "generate [both a functionality and a user interface] immediately and concurrently without any modification of software by a user	"dynamically [re-]generate[d, ing]": "generate or update when needed"

122. The parties disputes with respect to "dynamically generating" are (i) whether there is any functional limitation to "dynamically generating," and (ii) the scope of the temporal limitation imposed by that term. Consistent with the specification and prosecution history, Salesforce's construction does not permit any modification of the software (such as reprogramming, recoding, re-keying, and/or reformatting) by a user during this generation, while

AIT's construction improperly permits user intervention during generation including through reprogramming or recoding.

123. One of ordinary skill in the art at the time of the claim invention would have understood that the plain meaning of "dynamically" required at least generation of the UI and functionality "immediately and concurrently." Contemporary technical dictionaries confirm this understanding, requiring, e.g., "Occurring immediately and concurrently." Ex. 15 at 165 (Microsoft Computer Dictionary, 3rd Edition (1997)).

124. The specification requires that "dynamically generating" be construed to require that there be no modification of software by a user when generating an application. The specification states that earlier database software required "continual reprogramming" in order "to reflect a constant stream of changes" to regulations, and that "[t]his approach is not cost effective and, in effect, mortgages the database maintainer's future." Col. 8:2-8. The specification explains that the invention overcame these challenges by generating a user interface and functionality, in response to detected changes, without modification of software by the user. Col 8:36-44 ("These needs are met by the invention that, in one integrated system . . . converts the relevant changes into changes in work/task lists, data entry forms, [etc], ***without requiring the services of one or more programmers to re-program and/or recode the software items affected by the change***.") (emphasis added); Abstract 6-17 ("The system: … converts these changes into changes in data entry forms, data processing and analysis procedures, [etc.], ***without requiring the services of one or more programmers to re-key and/or reformat the items affected by the change***.") (emphasis added).

125. I note that AIT relies for its construction on the same dictionary definition that supports Salesforce's construction. That definition states: "Occurring immediately and concurrently. The term is used in describing both hardware and software; in both cases it describes some action or event that occurs when and as needed." The further phrase "when and as needed," on which AIT apparently bases its construction, in the context of this definition, simply reflects that the change has to take place right away.

126. AIT argues in its brief that Salesforce's proposed construction is flawed because "immediately and concurrently" introduces "temporal ambiguity" into the claims and "fails to identify 'immediately and concurrently' relative to what." (Br. at 17.) This criticism is misplaced: the remainder of the claim language specifies to what event the "immediately and concurrently" applies. For example, claims 1 and 21 of the '482 Patent makes clear that the application is dynamically generated [or re-generated] "when the client computer connects to the server computer."

127. By contrast, AIT's proposed construction equates "dynamically" with the isolated phrase "when needed," a term that on its own has no ascertainable boundaries and runs contrary to the specification and prosecution. The word "dynamically" does not appear in the specification, and the only reference to "when needed" in the specification sheds no light on its scope:

> The Java data management layer the end user sees is defined only by the metadata and is generated ***as needed*** by a single program that interprets what a form will look like.

128. (Br. at 21.) Here, neither the claims nor specification clarify whose need directs or causes generation, failing to provide any guidance as between the user, system, or a more generalized imperative to update an application, thereby rendering the asserted claims indefinite.

D. **"layer" / "portion of the server" or "portion"**

Term	Salesforce Proposed Construction	AIT Proposed Construction
"layer" '482 Patent, claims 1, 3, 5, 10, 20, 21, 23, 25, 30, 40.	Indefinite, or in the alternative, requiring at least "a group of data and/or functions that is separate and distinct from other such groups"	"a set of functionally or logically related software components"

Term	Salesforce Proposed Construction	AIT Proposed Construction
"portion of the server" or "portion" '111 Patent, claims 13-17.	Indefinite, or in the alternative, requiring at least "a subset of one server computer separate and distinct from other subsets"	"a functionally or logically related subset of one or more server computers"

129. AIT's proposed constructions of "layer" and "portion of the server" run counter to the claim language, specification, and prosecution history, and should be rejected. Further, AIT's constructions introduce ambiguity into these terms that would render the claims indefinite.

130. As an initial matter, one of ordinary skill in the art at the time of the claim invention would have understood that the plain meaning of "layer" required a group of data and/or functions that is separate and distinct from other such groups. Contemporary technical dictionaries confirm this understanding, requiring:

- "layer" - One of the divisions within ***which components or functions are isolated in computer system*** with layered architecture or a communications system with layered protocols. (McGraw-Hill dictionary of Scientific and Technical Terms (5th Ed. 1994).
- "layered architecture" - A technique used in designing computer software, hardware, and communications in which system or network components are ***isolated in layers*** so that changes can be made in one layer without affecting the others. (McGraw-Hill dictionary of Scientific and Technical Terms (5th Ed. 1994).
- "layer" - 1. The protocol or protocols operating at a particular level within a protocol suite, such as IP within the TCP/IP suite. Each layer is responsible for providing ***specific services or functions*** for computers exchanging information over a communications network (such as the layers in the ISO/OSI reference model) and ***information is passed from one layer to the next***. Although different suites have varying numbers of levels, generally the highest layer deals with software interactions at the application level, and the lowest governs hardware-level connections between different computers. See the table. See also ISO/OSI reference model, protocol stack, TCP/IP; 2. In communications and distributed processing, ***a set of rules and standards that handles a particular class of events***. (Microsoft Computer Dictionary (3d ed. 1997).)

- "layering" - Layering is a technique to write complex software faster and more easily. Layering is often used with public, open software. ***The idea is to have layers of software on top of other layers. Each performs a specific task.*** The idea is that if your software works at one layer — i.e. conforms to the rules of that layer — it should be compatible (i.e. work with) the layers of software above and below it. The most famous layered software is the seven-layer OSI (Open Systems Interconnection) model. It breaks each step of a transmission between two devices into a discrete set of functions. These functions are grouped within a layer according to what they are meant to accomplish. The data link layer, for example, is concerned with the transmission of frames of data between devices and covers protocols that are aimed at packaging raw data characters into frames, detecting and correcting errors when frames get lost or mutilated, arranging for retransmission and adding flags and headers so that DTE can recognize the beginning and end of a frame. ***Other layers serve other purposes. Each layer communicates with its counterpart through header records.*** The flexibility offered through the layering approach allows products and services to evolve. Accommodating changes are made at the layer level rather than having to rework the entire OSI model. Another layered software architecture is Microsoft/Inters Windows Telephony. It has three layers. At the lowest is SAPI, which is the Service providers' API. In the center is the actual Windows Telephony code. At the top is the TAPI — Telephony applications API. (Newton's Telecom Dictionary (2000).)

131. The claim language also supports Salesforce's proposed construction, as each claimed layer/portion has distinct functions and is recited separately, for example:

> 13. A system, comprising:
>
> a server accessible by a browser executed on a client device, the server including a first portion, a second portion, a third portion, and a fourth portion,
>
> the **first portion of the server** having information about unique aspects of a particular application,

> the **second portion of the server** having information about user interface elements and one or more functions common to various applications, the various applications including the particular application,
>
> the **third portion of the server** being configured to dynamically generate a functionality and a user interface for the particular application, the functionality and the user interface of the particular application being based on the information in the first portion of the server and the information in the second portion of the server, the third portion of the server being configured to send the functionality and the user interface for the particular application to the browser upon establishment of a connection between the server and the client device,
>
> the **fourth portion of the server** being configured to automatically detect changes that affect the information in the first portion of the server or the information in the second portion of the server.

'111 Patent, claim 13.

132. As recited by claim 13, the "first" portion/layer stores "information about unique aspects of a particular application," the "second" portion/layer stores "information about user interface elements and one or more functions common to various applications," the "third" portion/layer incorporates all changes to the information in the first and second portions/layers and generates the application (*i.e.*, the Java data management layer), and the "fourth" portion/layer detects external material changes that need to be incorporated into the metadata (*i.e.*, the change management layer). The claim language thus provides no suggestion of any overlap between layers/portions, but instead supports the separateness and distinctness provided by Salesforce's construction.

133. The specification also supports Salesforce's constructions of portion/layer. Notably, the Specification defines the "invention" in terms of the relationship of the layers / portions as set forth in Figure 1. Col. 8:50-51. Notably, these relationship of these layers shows them as separate and distinct from one another:

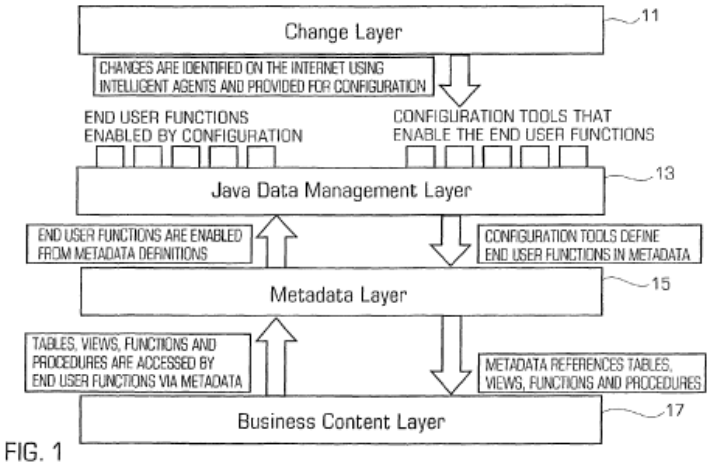


FIG. 1

134. In defining the features of the "invention" in terms of the relationship of the layers of Fig. 1, this description should limit the scope of the invention. Consistent with this disclosure, the remainder of the specification also separately describes the distinct functions performs by each layers of the patented invention; AIT provides no evidence that layers may functionally or physically overlap. Col. 12:15 - Col. 16:60.

135. AIT's citations to the specification purportedly demonstrating that layers are "interrelated" or "interconnected" do not advance AIT's construction of layer. (Br. at 15.) At most, the cited passages demonstrate only that metadata in the metadata layer define or reference information in the business content layer—for example, to enable management by the data management layer—not that these layers are indistinct or overlap.

136. I note that the patentees confirmed during prosecution that the claimed layers are separate and distinct. Specifically, the Examiner rejected over 60 pending claims as anticipated by U.S. Patent No. 5,960,200 ("Eager"). '482 Patent File History, 1/18/2006, Final Office Action. The patentees sought to overcome this rejection by arguing that the Examiner improperly relied on the same group of functionality in Eager (labeled 130) to meet both the claimed change detection and change incorporation layers:

> How can Eager's functionality later 130 be both "a third layer associated with the server computer that retrieves the data in the first and second layers in order to generate the functionality and user interface elements of the application" and "a change management layer for automatically detecting changes that affect an application"? **At most, it can be one of those layers, not both of them together.**

Ex. 5 at 14 ('482 Patent File History, 5/18/2006 Amendment and Remarks (emphasis original)).

137. The Examiner issued his notice of allowance for the '482 patent on December 28, 2007. In my view, the patentees thus disclaimed AIT's proposed construction as to "layer" and "portion," requiring that they be separate and distinct. The Court should therefore reject AIT's proposed construction.

138. Under AIT's proposed construction, the claims would permit an unspecified degree of physical overlap and functional redundancy between layers and portions. That is, two separately claimed layers/portions would be permitted to share any number of the same components, or even all of the same components, so long as there is some relationship, functional or logical, within a given layer. The patents-in-suit, however, provide no guidance as to any permissible overlap between various layers/portions, or how this would be technically implemented, and AIT identifies none in its brief. Instead, as set forth in more detail below, the intrinsic record underscores the physical and functional distinctness of the claimed layers.

139. Indeed, the distinction between the claimed layers is a necessary component to implement the invention, as overlap would prevent the system from operating as a "closed loop" that identifies regulatory and other changes on the Internet, incorporates those changes into the metadata underlying the system, and then generates the application from this updated metadata.

140. The ambiguity of the "layer" and "portion of the server" terms is compounded by the fact that individual examples of each (i.e., the first and second portions / layers) are not well grounded in the specification. Here, there is no apparent correspondence between the first and second layers required by the claims—which store metadata regarding features of the application that are "unique" and "common," respectively—and the "Business Content Layer" and "Metadata Layer" described by the specification.

141. Thus, under AIT's proposal, the specification fails to "provide objective boundaries for those of skill in the art," rendering these terms indefinite. That is, under AIT's proposal, the specification fails to "provide objective boundaries for those of skill in the art," rendering these terms indefinite.

142. To the extent that the intrinsic record sheds light on these claim limitations, the claim language, specification, and prosecution history confirm that "layers"/"portions" must at least be separate from each other and have distinct functions, consistent with Salesforce's proposed construction.

143. In Paragraph 35, Mr. Rosenberg argues that layers have very few requirements but rather just that they must be related software components: "A layer is generally understood to be a logical structuring mechanism for the elements that make up a software system." Yet, Mr. Rosenberg cites to industry websites that give technical definitions of layers that are actually much more consistent with Salesforce's proposed construction. Specifically, these websites describe layers in a way that require the functions of each layer to be separate and distinct from the code in other layers. The cited Microsoft website (https://msdn.microsoft.com/en-us/library/ff646997.aspx?f=255&MSPPError=-2147217396) explains that one of the values of layered software architectures is that the software can be split up at the layer boundaries and deployed on different servers ("Although it is very hard to distribute a single-layered application across multiple servers, it is much easier to divide the application at layer boundaries and distribute the different parts to multiple servers.") The reason this is possible is precisely because each layer is separate and distinct from each other. If there were some code that was shared across layers, it wouldn't be possible to deploy each layer on a different server.

144. Similarly, the other cited website (https://www.techopedia.com/definition/2016/layer-object-oriented-design) defines layers based on which modules they import. Specifically, the website defines layers to be "a set of classes that share the same module dependencies with other modules." Thus, according to this definition, all code within a layer must have the same module dependencies and code in different layers must have different dependencies. It is impossible for any two sets of code to have both the same and different module dependencies at the same time, and thus, given these constraints, two sets of code can not be within one layer, and split across two layers simultaneously. It is similarly impossible for one set of code to have one module dependency and a different module dependency at the

same time, and thus it would be impossible for one set of code to be part of two different layers. Thus, the very definitions that Rosenberg himself relies on to define layers explain that layers must be separate and distinct.

E. **Unique / Common (‘111 patent, claim 13; ‘482 patent, claims 1, 21)**

Term	Salesforce Proposed Construction	AIT Proposed Construction
“unique aspects” / “information about [the] unique aspects of a particular application” ‘111 patent, claim 13. ‘482 claims 1, 21.	Indefinite	Subject to constructions above, no construction necessary – plain and ordinary meaning..

Term	Salesforce Proposed Construction	AIT Proposed Construction
“information about user interface elements and one or more functions common to various applications” ‘111 Patent, claim 13. ‘482 Patent, claims 1, 21.	Indefinite, or in the alternative, requiring at least “information about user interface components and functions used by multiple different applications, excluding any unique aspects of those applications”	“metadata that defines user interface elements and/or application functions common to multiple applications”

145. I note that the asserted claims require a first metadata layer with “information about unique aspects of a particular application” and a second metadata layer with “information about user interface elements and on or more functions common to various application.” The patents provide no meaningful distinctions or boundaries as to when an aspect of an application is “unique” or “common,” thereby rendering the asserted claims indefinite. Neither the claims, specification, nor prosecution history provide any guidance as to when aspects of an application are unique or common, and AIT identifies no guidance in its brief.

146. Compounding this lack of guidance, the specification characterizes the patented invention as having only a single metadata layer, not separate metadata layers for unique and common metadata, respectively:

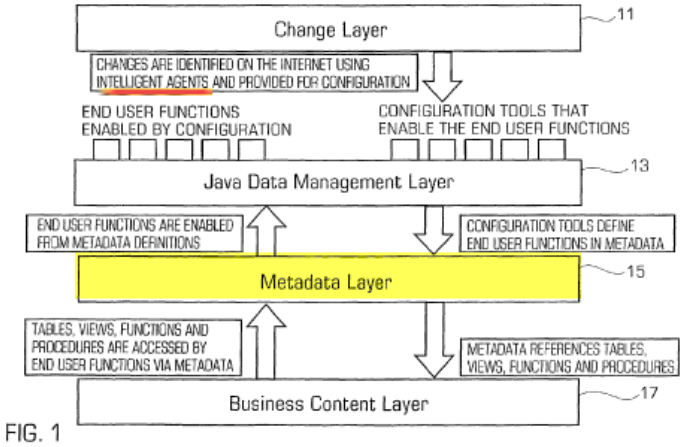


FIG. 1

147. Given this lack of guidance, the boundaries and scope of "unique" and "common" are difficult to ascertain. As to uniqueness, it is unclear whether absolute uniqueness from all other applications is required, or whether uniqueness is only required with respect to some subset, for example, applications used by employees of a particular company.

148. Similarly, as to commonness, the claim language requires only that common elements must be shared by "various applications," but it does not specify whether two, twenty, or two million applications would be sufficient, and whether this commonness may be incidental to application design, i.e., a byproduct of customization. In particular, the intrinsic records provides no guidance on whether common user interface elements or functions may result from customization and, if so, how such functions are distinct from unique functions. Here, at best, the distinction between unique and common is difficult to ascertain, and the lack of sharp distinctions renders these claim terms indefinite.

F. **Business Content Database**

Term	Salesforce Proposed Construction	AIT Proposed Construction
"business content database" '482 Patent, claims 3, 23.	Indefinite	"a data store containing data specific to particular business operations"

149. AIT's proposed construction of this term, and the use of this claim term in dependent claims 3 and 23, illustrate how the this claim term fails to inform, with reasonable certainty, those skilled in the art about the scope of the invention." For example, in claim 23, it is the "first layer," with information regarding "unique aspects of a particular application," that comprises the "business content database." Thus, for the same reasons as set forth in connection with the "unique / common" limitations above, the "business content database" claim term is indefinite.

150. AIT's proposed construction, which recites that the business content database "includes data specific to a particular application," compounds the ambiguity of this claim term. The specification does not use the term "data specific" to an application, nor does it provide any support for incorporating such a limitation. Moreover, one of skill in the art would further be unable to determine, with reasonable certainty, whether "data specific to particular business operations" corresponds to "unique aspects of a particular application" or "information … common to a variety of applications." For this additional reason, this claim limitation is indefinite.

G. **"logical design" / "physical design" / "physical structure"**

Term	Salesforce Proposed Construction	AIT Proposed Construction
"logical design" '482 Patent, claim 24. '111 Patent, claim 15.	"an arrangement of data in a series of logical relationships referred to as entities or attributes"	"an abstract representation of the data flows, inputs, and outputs of an application"
"physical design" '482 Patent, claim 24. '111 Patent, claim 15.	"description of a physical database including tables and constraints"	"the input and output processes of an application"
"physical structure"	"structure of a database that can be seen and operated on by the	"the components, their relationships and arrangements, that form an

'482 Patent, claim 24. '111 Patent, claim 15.	operating system, such as the physical files stored on a disk"	application"

151. I understand that the terms "logical design," "physical design," and "physical structure" appear in a single asserted dependent claim of each of the patents-in-suit:

> **'482 patent, claim 24:** The method of claim 23, wherein the data further comprises one or more of business knowledge**, logical designs, physical designs, physical structures** and relationships associated with the predetermined business application.
>
> '**111 patent, claim 15:** The system of claim 13, wherein the information of the first portion of the server includes at least one of business knowledge**, logical designs, physical designs, physical structures**, and relationships associated with one or more predetermined business applications.

152. The parties' principal dispute here is whether the terms "logical design," "physical design," and "physical structure" describe attributes of databases instantiated by an application (Salesforce's proposed construction) or attributes of the application itself (AIT's proposed construction). Because the claim language, specification, and pertinent extrinsic evidence confirm that these terms describe the design and structure of databases instantiated by an application, the Court should adopt Salesforce's proposed construction.

153. The claim language confirms that "logical design," "physical design," and "physical structure" are types of data that can be stored in a "business content database." In particular, claim 24 depends from claim 23, which states that the "first layer comprises a business content database having data about one or more different predetermined business applications." Claim 24 states that "logical design," "physical design," and "physical structure" refer to types of data that can be stored in this business content database.

154. The specification indicates that these three types of data characterize the design and structure of databases instantiated by an application. The specification states that these three types of data, along with "business knowledge," are stored in a business content layer. This business content layer is "defined by and referenced in" metadata stored in an Oracle or similar database system (the metadata layer) that permits the patented invention to instantiate a database using the

"logical designs," "physical designs" and "physical structure" that define it, i.e., "so that the necessary objects, tables, columns, relationships, functions, procedures and data can be read and updated by the Java data management layer that permits the instantiation of database." Col. 12:17-28.

155. The extrinsic evidence, excerpted from Oracle database manuals, further confirm that "logical designs," "physical designs" and "physical structure" define the design and structure of databases instantiated by an application. Ex. 18 at 2-2 ("A logical design is a conceptual, abstract design. You do not deal with the physical implementation details yet; you deal only with defining the types of information that you need. The process of logical design involves arranging data into a series of logical relationships called entities and attributes."); Ex. 18 at 3-2 ("In a sense, logical design is what you draw with a pencil before building your warehouse and physical design is when you create the database with SQL statements. During the physical design process, you convert the data gathered during the logical design phase into a description of the physical database, including tables and constraints.").) Notably, the embodiment of the invention described in the patents-in-suit, and in particular the business content database, are implemented using an Oracle database system. (Col. 12:31-32; 16:61-65.) Thus, Salesforce's proposed construction of these terms should be adopted.

H. **Builder Module**

Term	Salesforce Proposed Construction	AIT Proposed Construction
"builder module" ('482 claim 10)	"self-contained unit of software capable of generating part of an application"	"a software tool to construct an application or part of an application from metadata"

156. In my view, one of ordinary skill in the art would understand that "module" has a particular meaning in the art, i.e., it is "self contained unit of software." Ex. 12 at 418 (Comprehensive Dictionary of Electrical Engineering); Ex. 14 at 355 (Webster's New World Dictionary of Computer Terms); Ex. 15 at 313 (Microsoft Computer Dictionary 3d ed.). AIT's construction ignores this plain meaning, relying instead on using the expression software "tool." However, the term "tool" itself is not one that is readily understandable by a lay jury, and thus AIT's construction is likely to add confusion.

157. I declare under the penalty of perjury under the laws of the United States of America that the foregoing is true and correct.

Executed in Takoma Park, Maryland, on October 16, 2015.

Benjamin B. Bederson